

## CLAIMS

What is claimed is:

1. A system for enabling the exchange of data between at least one remote user having a client application program running in an off-line mode and a server communicatively coupled to the application program over a network, said system comprising:
  - a client logical module adapted to:
    - receive user credentials;
    - detect the generation of an outgoing message generated through the application program;
    - transfer the outgoing message over the network to a domain module;
    - receive incoming messages from the domain module; and
    - transfer the incoming messages to the application program while the application program is operating in an off-line mode;
  - a domain module communicatively coupled to the client logical module and operating in association with the server, said domain module being adapted to:
    - impersonate the remote user, thereby appearing to the server as though the application program is connected directly to the server in an on-line mode;
    - receive outgoing messages from the client logical module;
    - transfer the outgoing messages to the server;
    - receive a message from the server; and
    - provide the received messages to the client logical module in the remote client.
2. The system of claim 1, wherein the network is a TCP/IP network.
3. The system of claim 1, wherein the application program is an email application program.
4. The system of claim 3, wherein the email application program is OUTLOOK.
5. The system of claim 1, wherein the domain module is further adapted to impersonate the remote user by performing a login procedure on behalf of the application program, thereby opening a communication session between the client logical module and the server.
6. The system of claim 4, wherein the communication session between the the client logical module and the server is a MAPI session.

7. A method for exchanging data between a plurality of remote clients, each remote client running an email application program in an off-line mode, and a server in a domain which is connected to the remote clients over a TCP/IP network, said method comprising the steps of:

5 sending a login request from at least one of the plurality of remote clients to a domain module;

in response to receiving the login request, said domain module impersonating the remote client by logging into the server serving the remote client;

opening a communication session between the remote client and the server;

10 and

transferring messages between the remote client and the server via the domain module, whereby the email application program appears to operate as though it is on-line with the server.

8. The method of claim 7, wherein the remote client includes a client logical  
15 module, and the step of sending the login request further comprises the steps of:

receiving credentials from the remote client; and

the client logical module forwarding the login request with the credentials to the domain module.

9. The method of claim 7, wherein the server is an exchange server.

20 10. The method of claim 7, wherein the communication between the domain module and the server is using MAPI.

11. A system for enhancing perceived throughput between a plurality of remote clients running an OUTLOOK application and an exchange server in a domain which is connected to the remote clients over a TCP/IP network, said system comprising:

25 a client logical module for each remote client, the client logical modules being adapted to:

receive credentials for a user of the remote client;

receive outgoing messages from a remote client outbox and to transfer the outgoing messages over the TCP/IP network to a domain module; and

30 receive messages from the domain module and transfer them to a remote client inbox in the OUTLOOK application while the OUTLOOK application is operating in an off-line mode;

a domain module, which is connected between said TCP/IP network and the domain, for each said plurality of remote clients, said domain module being adapted to:

impersonate the remote client by spoofing the exchange server to operate as though the remote client is connected directly to the domain;

5 login into the exchange server using the credentials of the user of the remote client;

open a MAPI session for the remote client;

receive messages from the client module of the remote client;

transfer OUTLOOK messages to the exchange server;

10 receive messages destined to a remote client from the exchange server;

and

submit the messages to the appropriate client module of the destined remote client, whereby using said system allows the delivery of messages between the plurality of remote clients and the exchange server in an off-line mode of operation of the

15 OUTLOOK application without having to modify the exchange server.

12. A method for exchanging data between a remote client running an application program in an off-line mode, and a server operating within a domain to which the remote client is communicatively coupled, said method comprising the steps of:

- receiving credentials for a user of the remote client;
- 5 detecting a message from the application program that is directed to the server;
- reformatting the message from MAPI format to a proprietary format;
- transferring the message to the server over a communication channel;
- detecting the reception of the message at the domain;
- 10 reformatting the reformatted message from the proprietary format to the MAPI format to create a MAPI message; and
- providing the MAPI message to the server.

13. The method of claim 12, further comprising the steps of:

- sending a login request to the server;
- 15 in response to receiving the login request, emulating the actions that would normally be taken by the application program to login to the server; and
- opening a MAPI session between the application program and the exchange server, whereby the application program appears to operate as though it is on-line with the server.

20 14. The method of claim 12, wherein a MAPI session exists between the application program running on the remote client and the server to facilitate communication of messages, further comprising the steps of:

- sending a disconnect request to the server;
- in response to receiving the disconnect request, emulating the actions
- 25 that would normally be taken by the application program to logoff the server; and
- closing the MAPI session between the application program and the exchange server.

15. The method of claim 12, further comprising the steps of:

- detecting a message from the server that is directed to the application
- 30 program;
- reformatting the message from MAPI format to a proprietary format;
- transferring the message to the application program over a communication channel;

detecting the reception of the message at the remote client;  
reformatting the reformatted message from the proprietary format to  
the MAPI format to create a MAPI message; and  
providing the MAPI message to the application program.

5           16. The method of claim 15, wherein the application program is an email  
application program and the step of providing the MAPI message to the application program  
comprises placing the MAPI message into the inbox of the email application program.

17. The method of claim 12, wherein the application program is an email  
application program and the step of detecting a message from the application program  
10       comprises detecting a new message in the outbox of the email application program.

18. The method of claim 17, wherein the email application program is  
OUTLOOK, further comprising the steps of:

receiving a profile selection, the profile selection being associated with  
enabling the off-line operation; and

15           enabling the off-line operation in conjunction with the profile selection.

19. The method of claim 17, wherein the email application program is  
OUTLOOK, further comprising the steps of:

receiving a profile selection, the profile selection not being associated with  
enabling the off-line operation; and

20           disabling the step of detecting a message from the application program that is  
directed to the server.